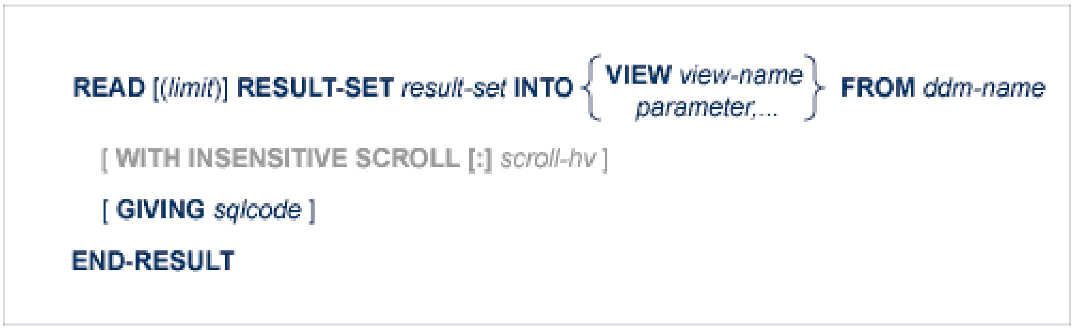
READ RESULT SET



```
READ [(limit)] RESULT-SET result-set INTO { VIEW view-name } FROM ddm-name
                                          { parameter,...  }
    [ WITH INSENSITIVE SCROLL [:] scroll-hv ]
    [ GIVING sqlcode ]
END-RESULT
```

Function

The statement READ RESULT SET is used to read a result set which was created by a stored procedure that was invoked by a previous CALLDBPROC statement.

The READ RESULT SET statement can only be used in conjunction with a CALLDBPROC statement.

As result-set you specify a result-set locator variable filled by a preceding CALLDBPROC statement. Result-set has to be a variable of format/length I4.

Note:
If a Syncpoint operation takes place between the CALLDBPROC statement and the READ RESULT SET statement, the result sets can no longer be accessed by the READ RESULT SET statement.

limit

You can limit the number of rows to be read. You can specify the limit either as a numeric constant (0 to 99999999) or as a variable of format N, P or I.

ddm-name

As *ddm-name* you specify the name of the DDM which is used to "address" the database executing the stored procedure. For more information, see ddm-name.

WITH INSENSITIVE SCROLL [:] *scroll_hv*

The WITH INSENSITIVE SCROLL [:] scroll_hv clause belongs to the SQL Extended Set. Using this clause causes the application to use an insensitive scrollable cursor to access the result set created by the previously invoked stored procedure. In order to use this clause, the stored procedure must have created the result set with a scrollable cursor. The scroll_hv has to be an alphanumeric Natural variable which contains the scrolling direction. The scroll_hv will be evaluated each time the READ RESULT SET processing loop is executed.

If the GIVING sqlcode option is specified as well, the processing loop will stay open, even if an sqlcode +100 (row not found) is returned from the RDBMS.

The processing will be terminated, if the application issues an ESCAPE statement or if the sqlcode +100 (row not found) is encountered five times successively without a terminal I/O.

If the GIVING sqlcode option is not specified , the processing loop will be closed, if any sqlcode other than 0 (successs) is returned from the RDBMS.

GIVING sqlcode

This option may be used to obtain the SQL code of the SQL "fetch" operation used to process the result set.

If this option is specified and the SQL code of the SQL operation is not "0", no Natural error message will be issued. In this case, the action to be taken in reaction to the SQL code value has to be coded in the invoking Natural object.

The *sqlcode* field has to be a variable of format/length I4.

If the GIVING *sqlcode* option is omitted, a Natural error message will be issued if the SQL code is not "0".

Example

See the example in the CALLDBPROC statement.

In addition, see the corresponding Natural database interface documentation in the Natural for Mainframes documentation.